

# ProtoDUNE FW-Hit-Finding Data Captures

---

19.8.20

Joel Greer

# Data Catalogue

Folder	Date	SSR	Pedestal Words	HF links active per super-logic region (max 5)	Board-reader version
N/A	4/6/20	NO	NO	1	fdaq?
APA6-wHits	24/6/20	YES	NO	1	fdaq/standalone?
APA6-newhits	6/7/20	YES	NO	1	standalone
APA6-newhits2	6/7/20	YES	NO	1	standalone
APA6-10072020 - testing?	10/7/20	YES	NO	5?	standalone
APA6-shortruns (3 captures) - testing?	10/7/20	YES	NO	5?	standalone
APA6-newFW-Nx (5 captures)	10/7/20	YES	NO	1,2,3,4,5	standalone
APA6-071620 (4 different channel masks)	16/7/20	YES	YES	1	standalone
APA6-071720 (8 different hf-link/threshold combinations)	17/7/20	YES	YES	1 or 5	standalone

# Data Catalogue

Kept track of in dataflow-firmware issue 82:

<https://gitlab.cern.ch/DUNE-SP-TDR-DAQ/dataflow-firmware/-/issues/82>

- SSR - firmware has state save/restore implemented
- Pedestal words - containing extra words for accumulator and median
- HF links active per super-logic-region (SLR) - max of 5 per SLR. All hit links in 1 SLR are written to a single binary file. 5 adc links are written to 5 binary files per SLR. 1 HF link active means only have to look at 1 adc link.
- Board-reader version - all useful captures use standalone BR

Current code able to look at captures with 1 or multiple HFs per SLR. Code needs small extension to deal with looking at multiple raw adc captures at once, for example for comparing hits from captures with 5 HF-links active with the raw adcs.

# Investigation of the Captures

- To histogram all hits in a capture I use a crawler to parse the binary files: `rawCrawlerHitBinaryCheck.py`
- To extract adc data need to use converted C++ tool: `extract_adcs_from_timestamp.cpp`
- To convert hit binary captures to 33b: `hitBinaryMangler.py`
- To convert 33b to hit objects and print to file: `hitFormatter.py`
- To run TPG on extracted adc data and compare to extracted hits: `hitComparison.py`
- To convert hit binaries to input format for event display: `evtdisplayhits.py`

Python tools are in dataflow-software:

[https://gitlab.cern.ch/DUNE-SP-TDR-DAQ/dataflow-software/-/tree/jgreer/tpg\\_pedchanfix](https://gitlab.cern.ch/DUNE-SP-TDR-DAQ/dataflow-software/-/tree/jgreer/tpg_pedchanfix)

C++ tools are here: <https://github.com/JatGreer/felix-long-readout-tools>

There are a few extra steps required before using some of these scripts, most commonly determining the timestamp range over which adcs and hits in captures overlap, which is manual and a bit convoluted

# Investigation of the Captures

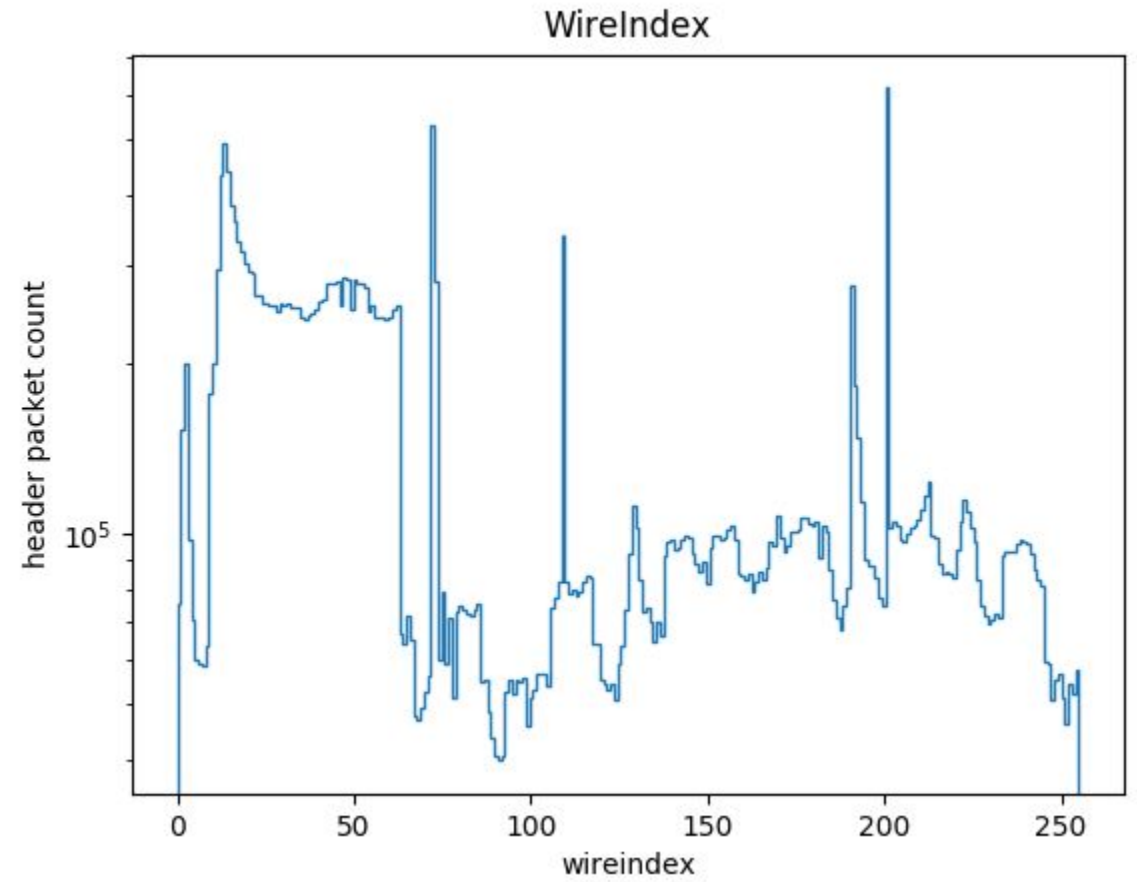
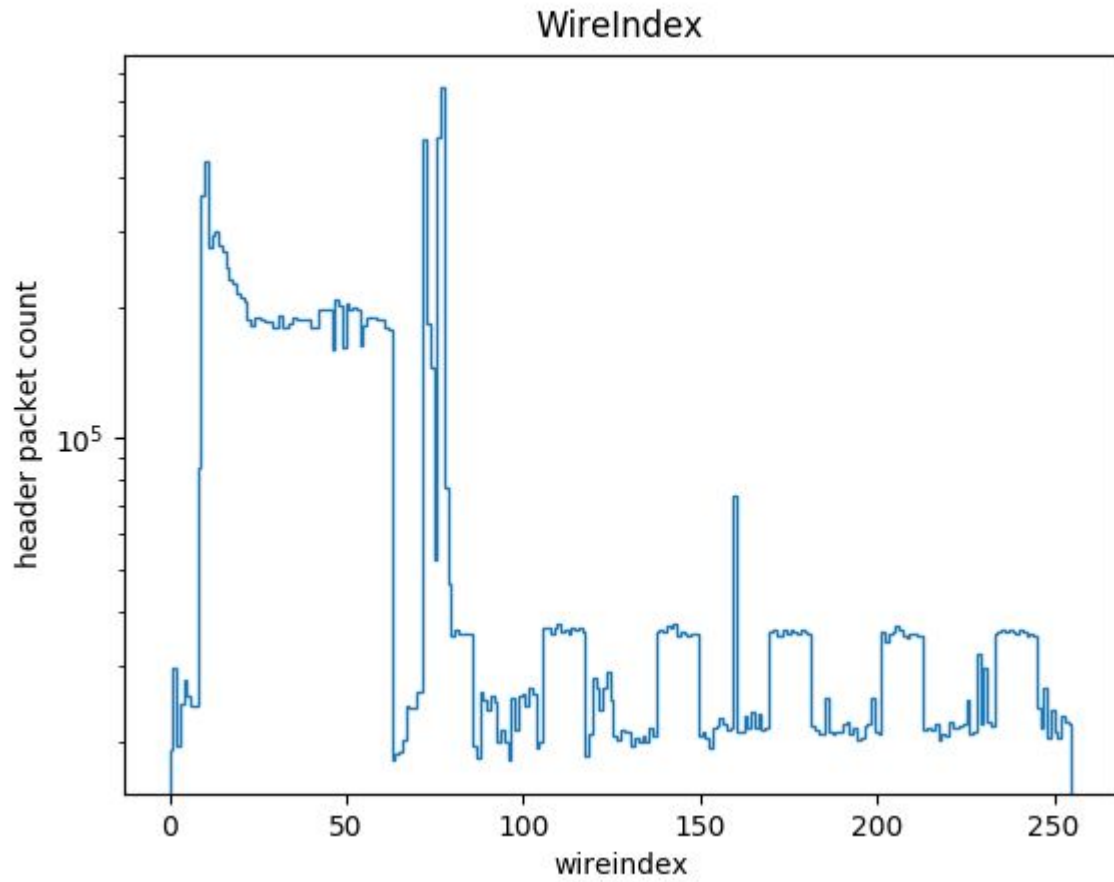
---

Looking at histogrammed hit quantities from:

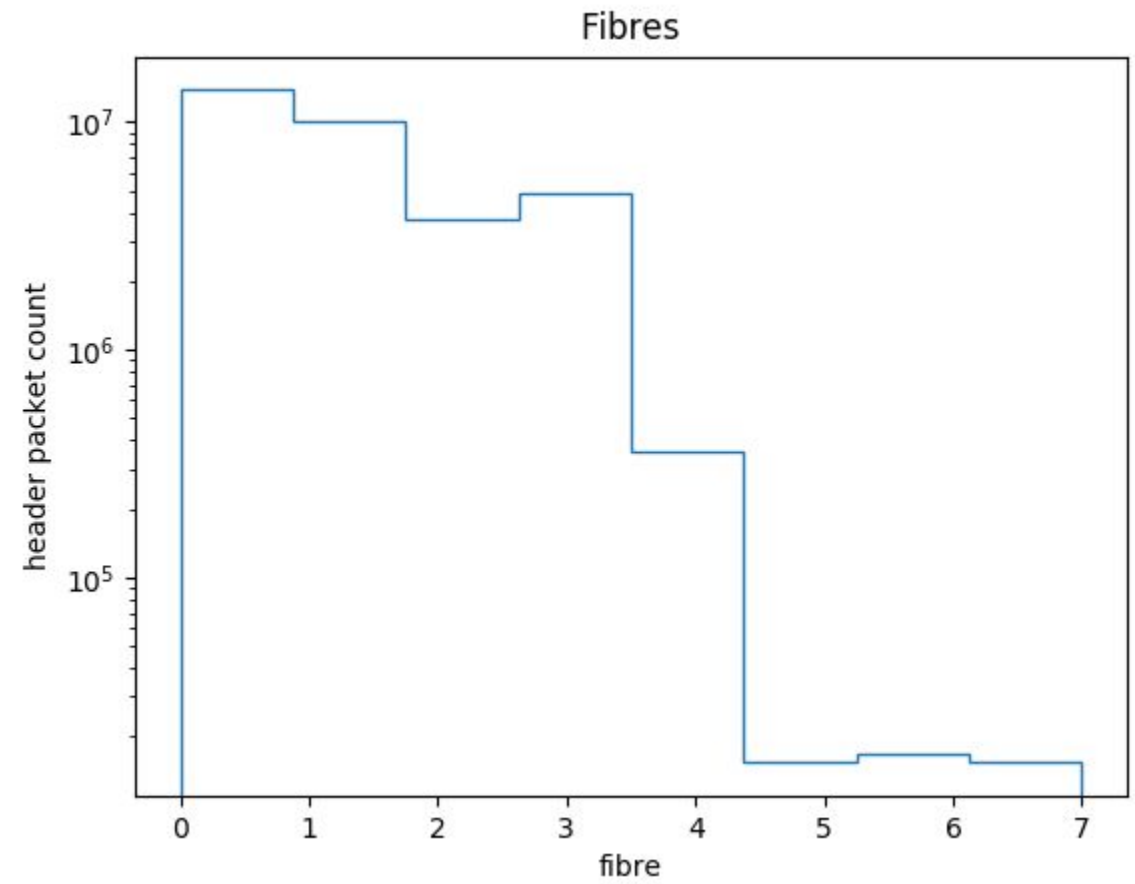
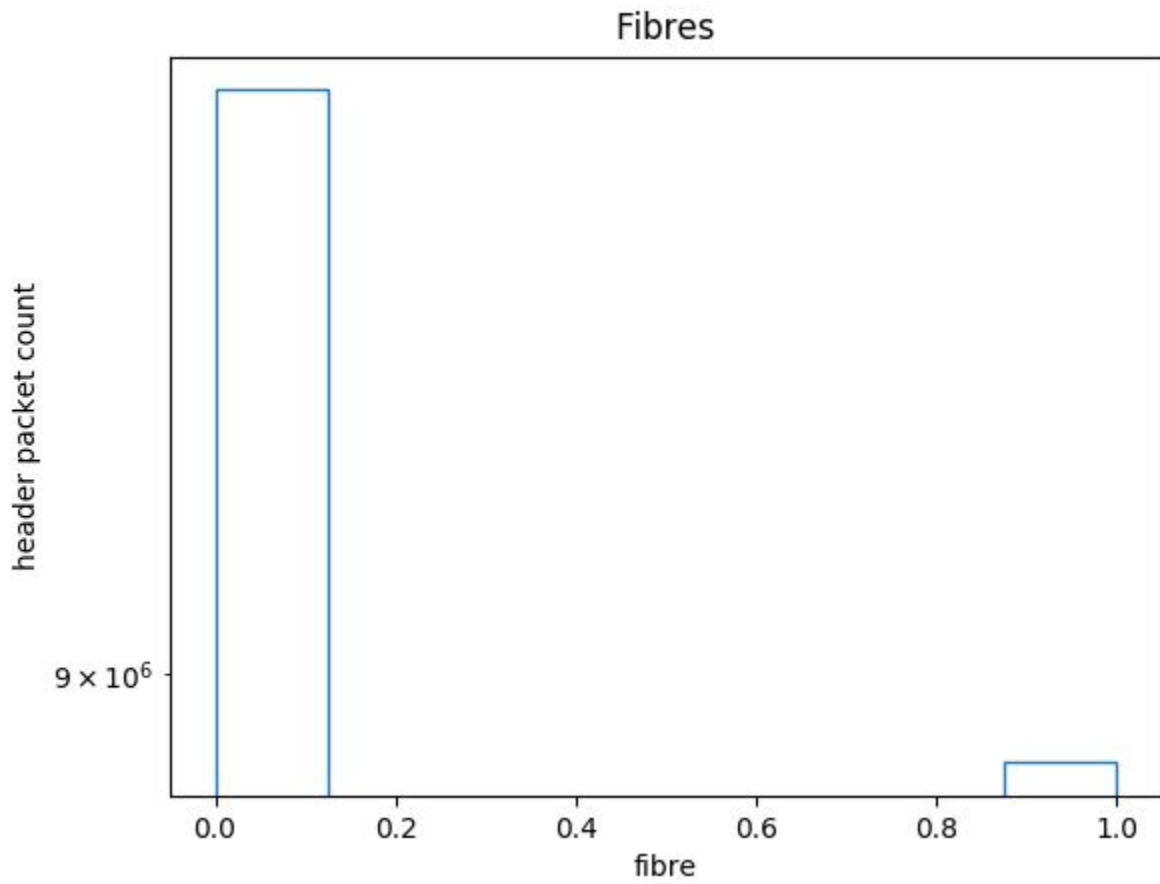
- APA6-newhits2 , plots on left
- APA6-071720, capture 8 (pedestal words, 10 HF links, threshold=70), plots on right

APA6-071720 allows us to see accumulator and medians as well but otherwise should be similar distributions of hit quantities.

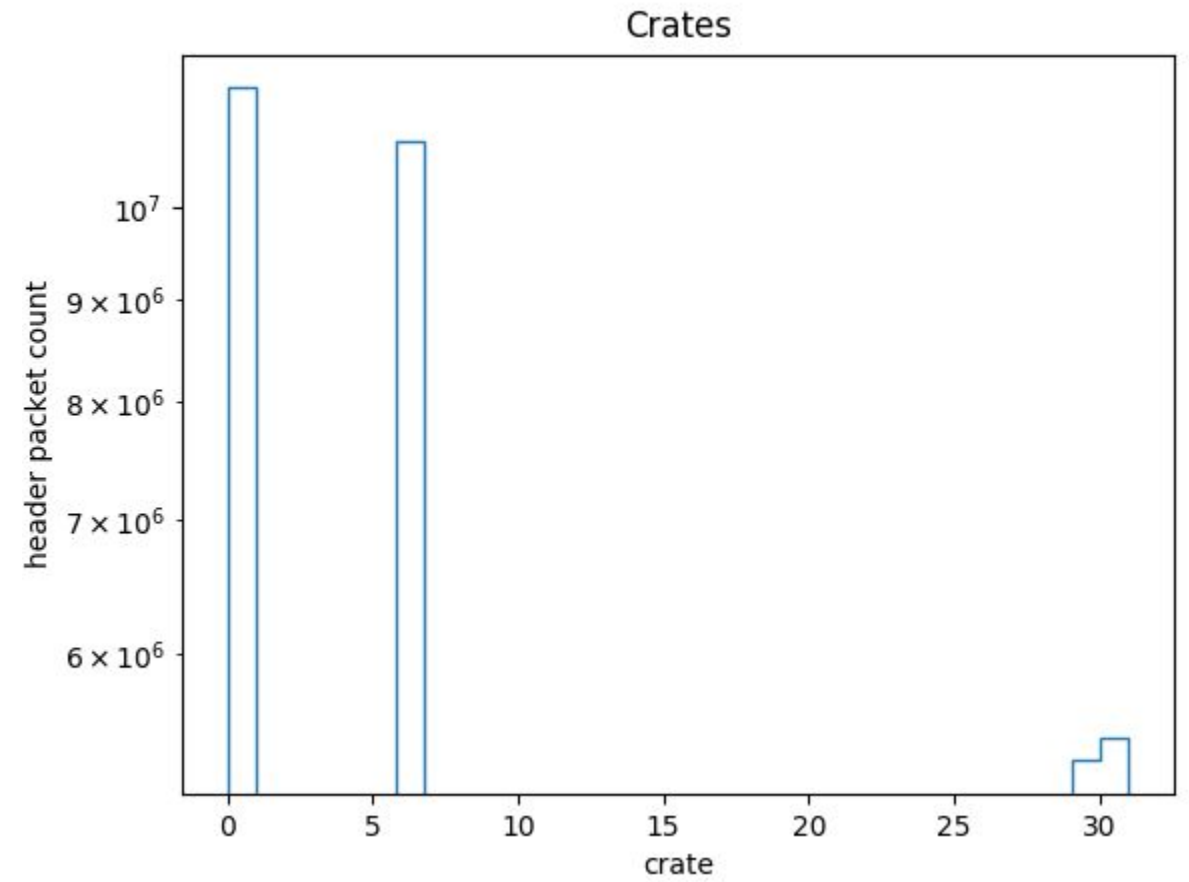
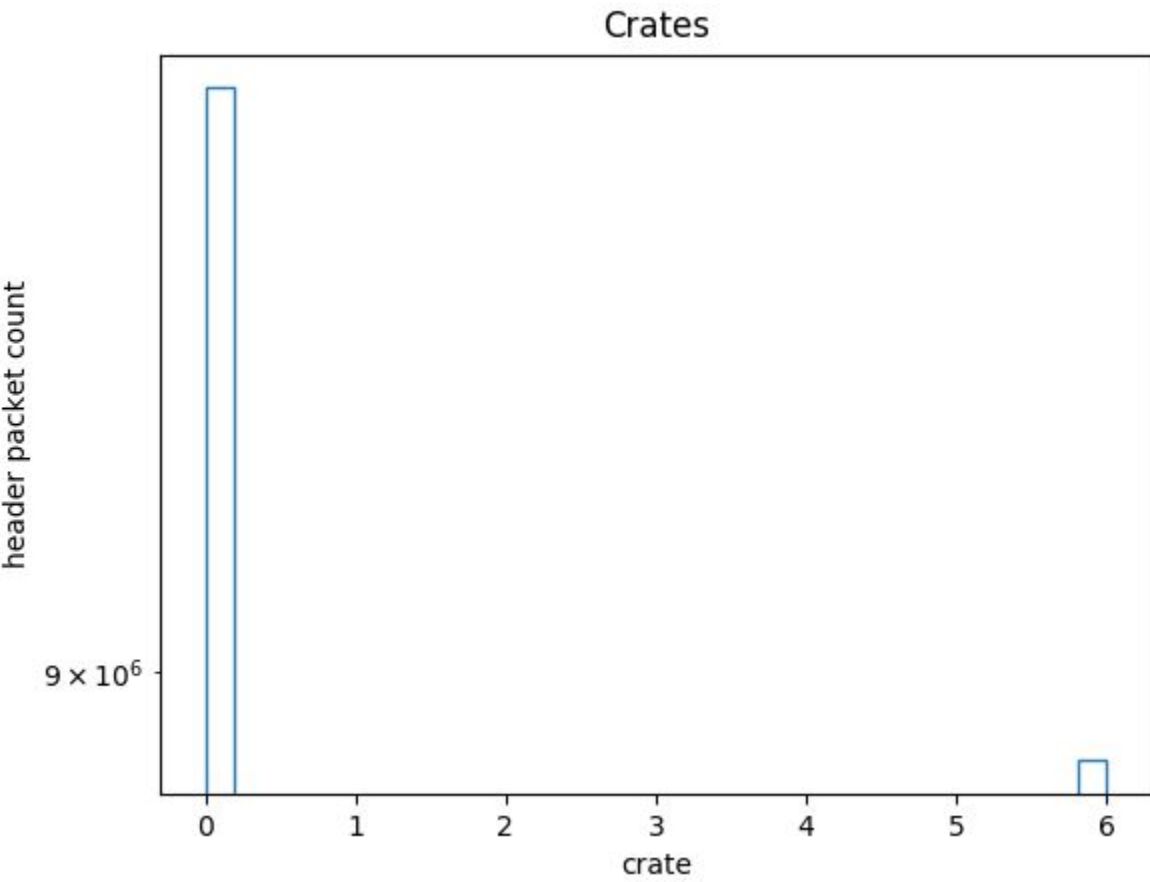
# WireIndex



# Fibre

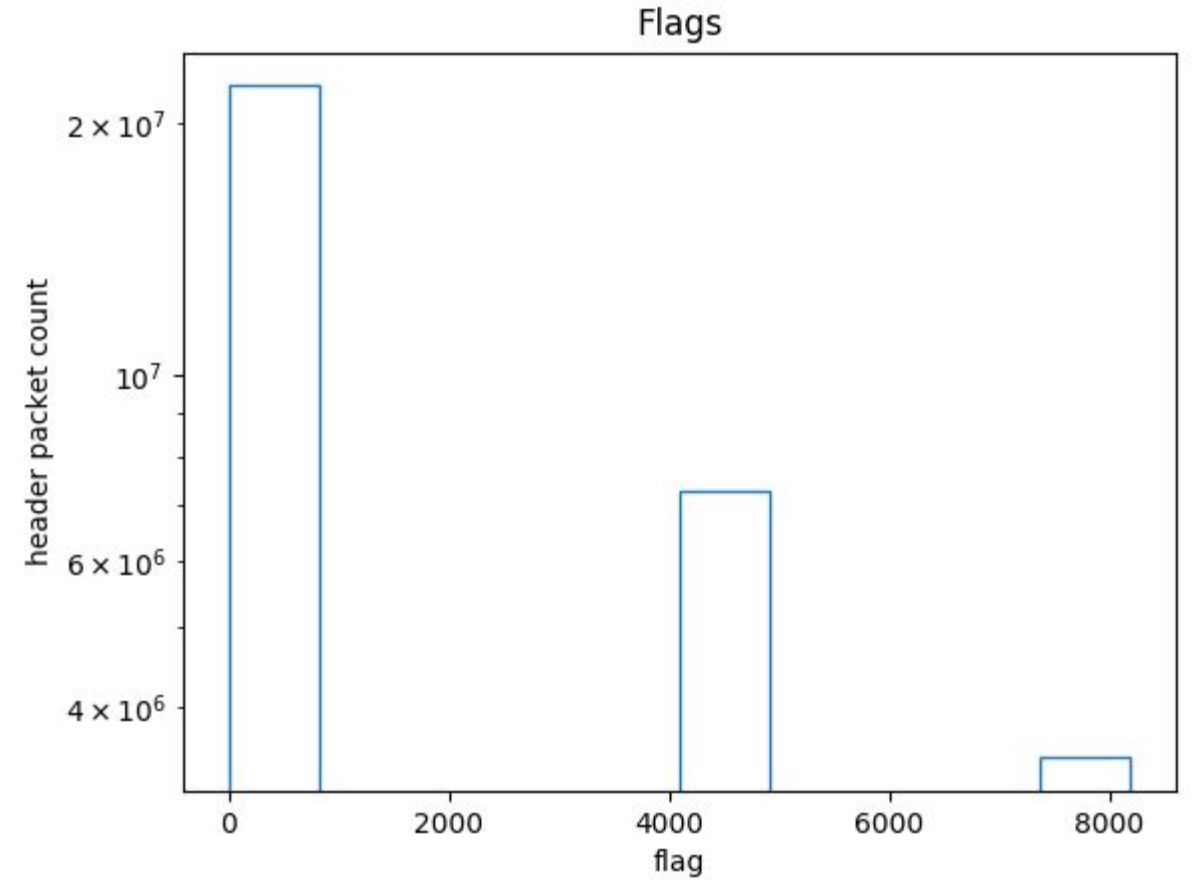
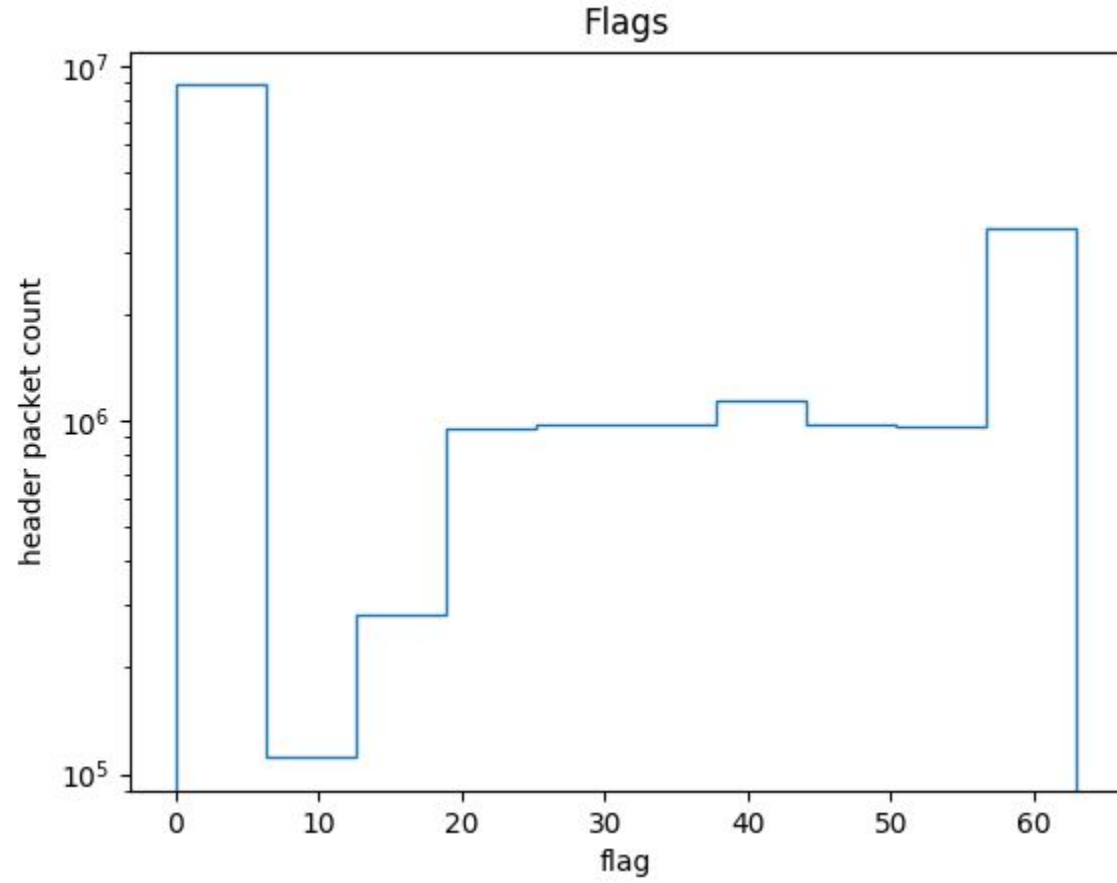


# Crate

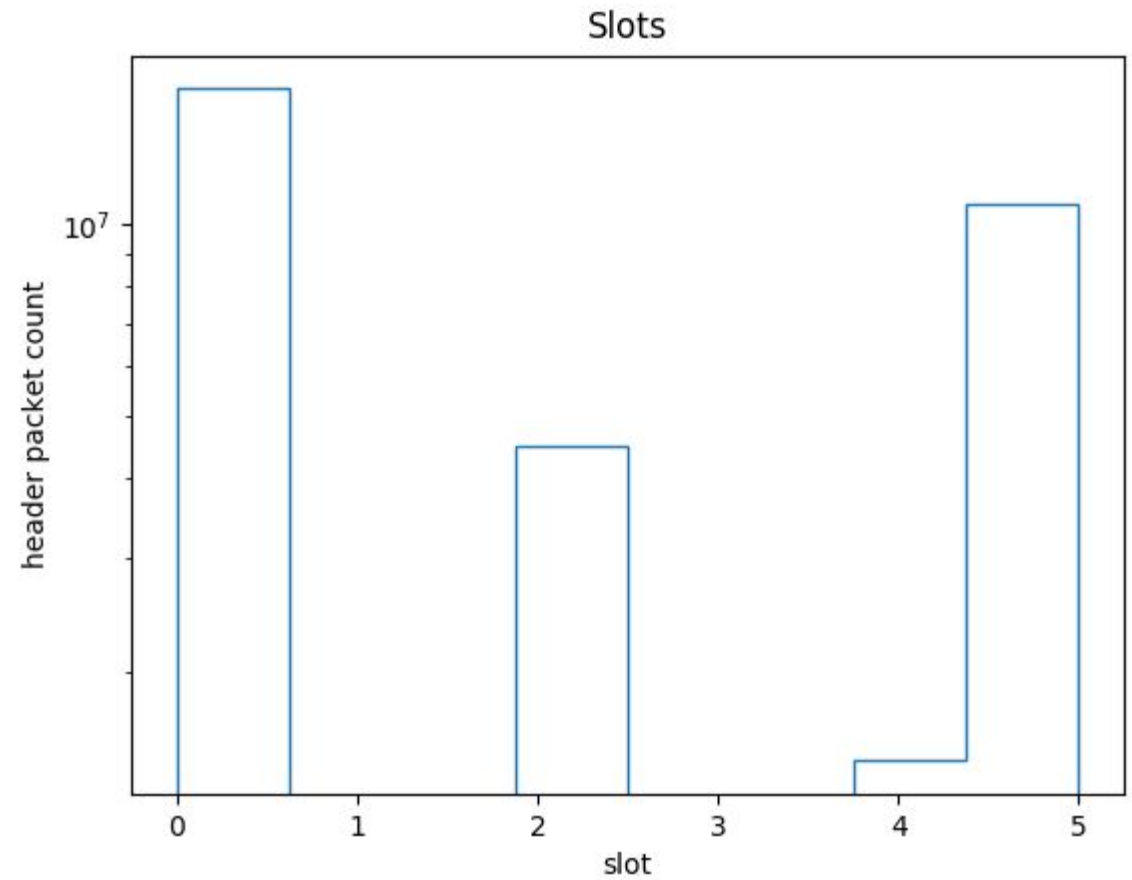
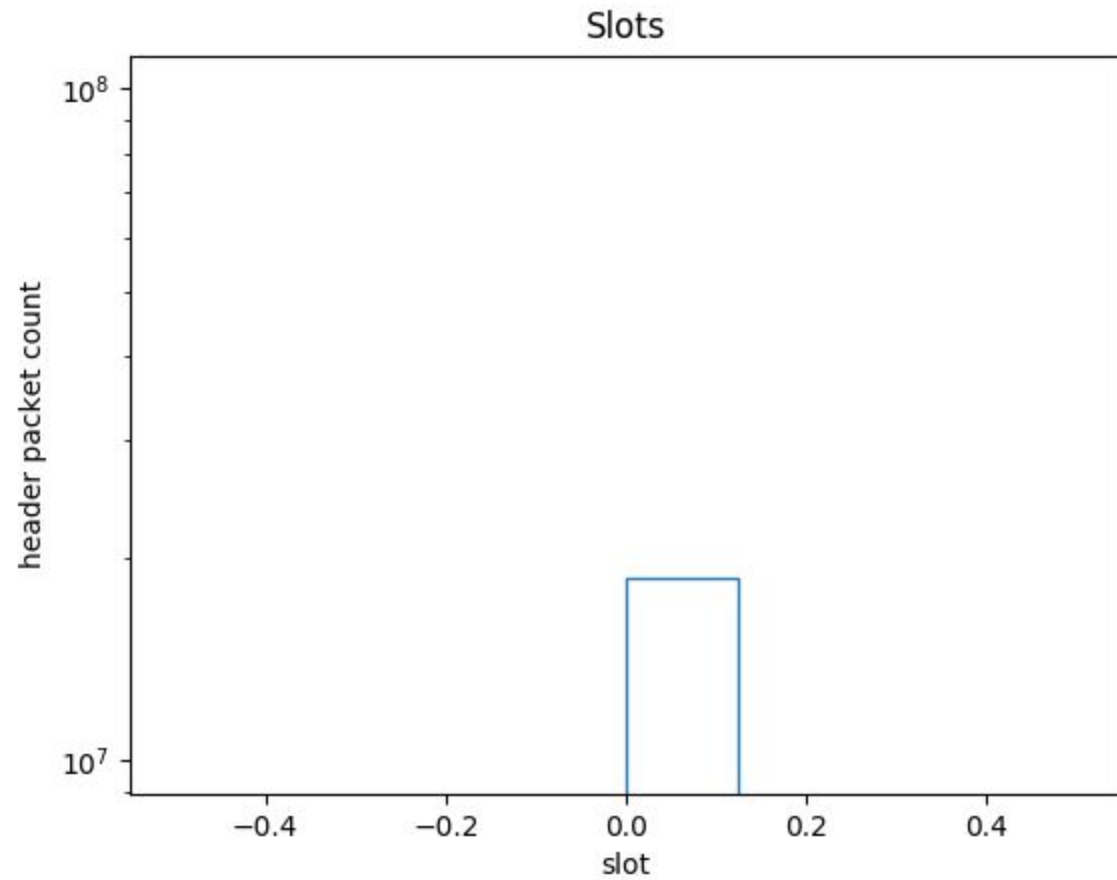




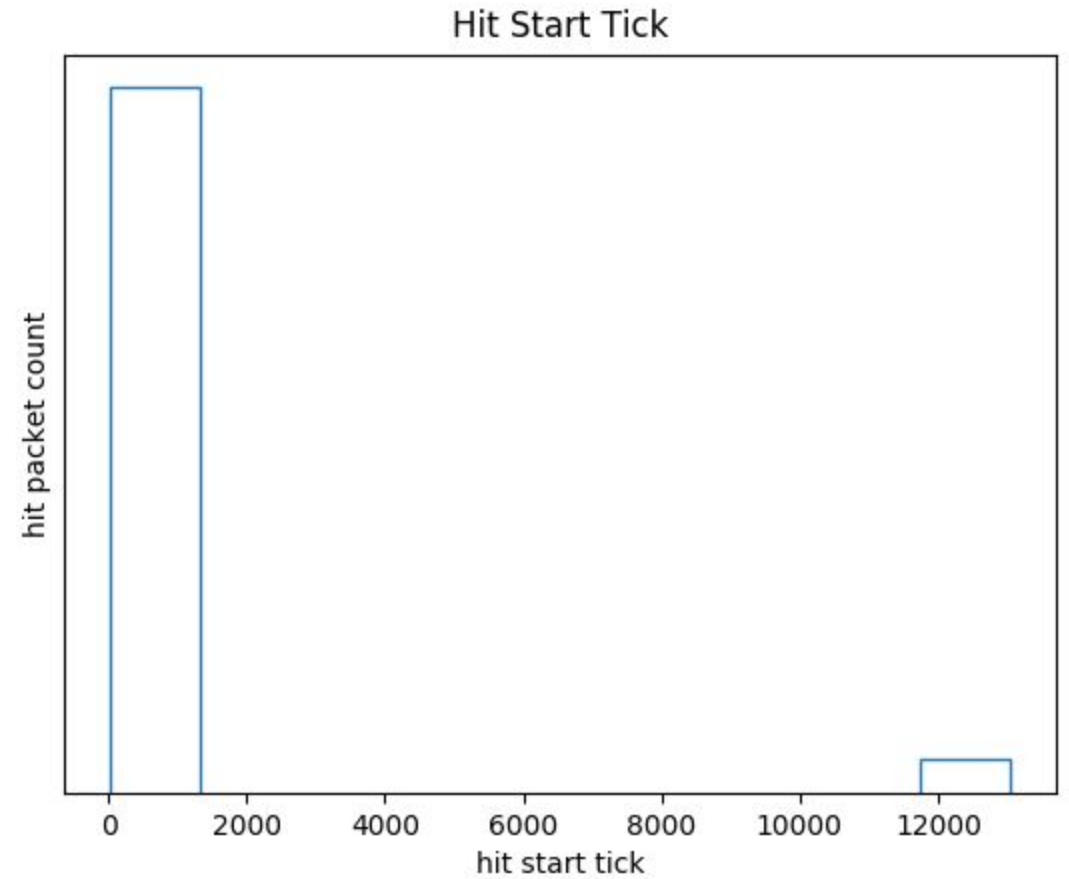
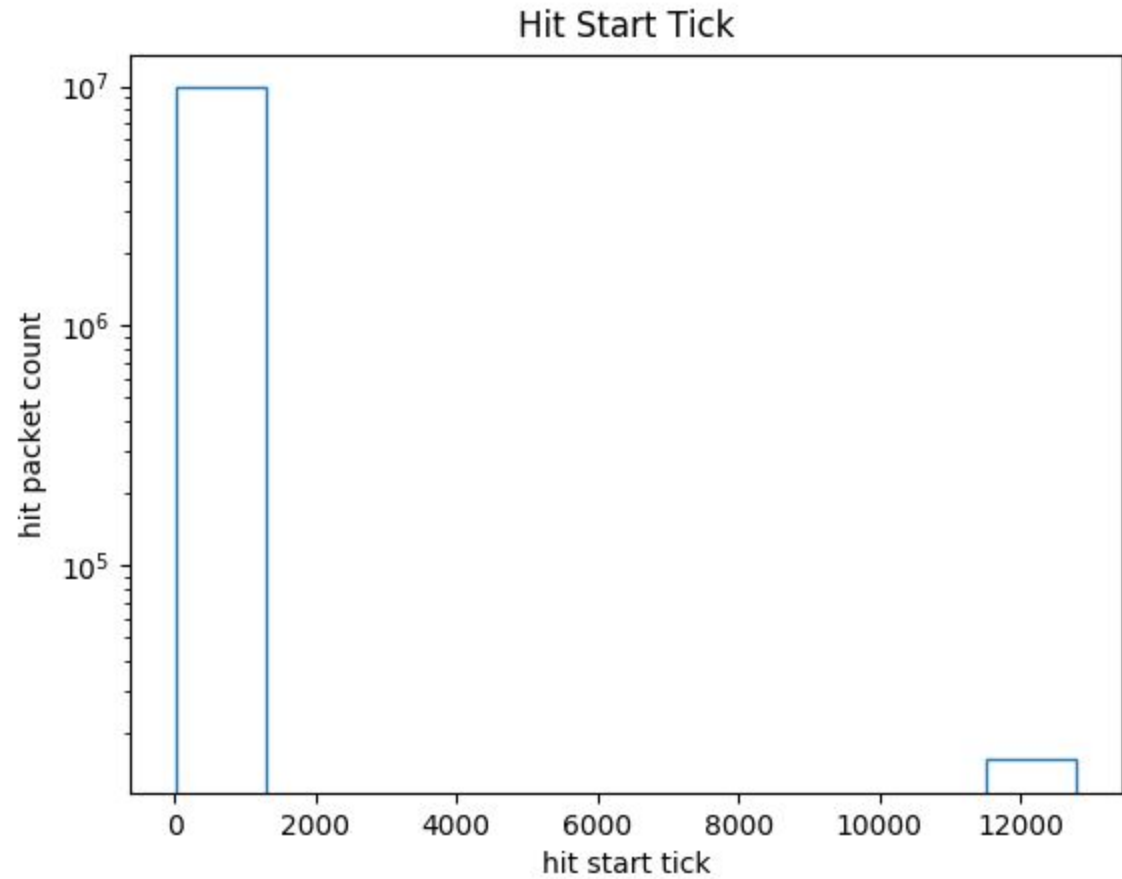
# Flags



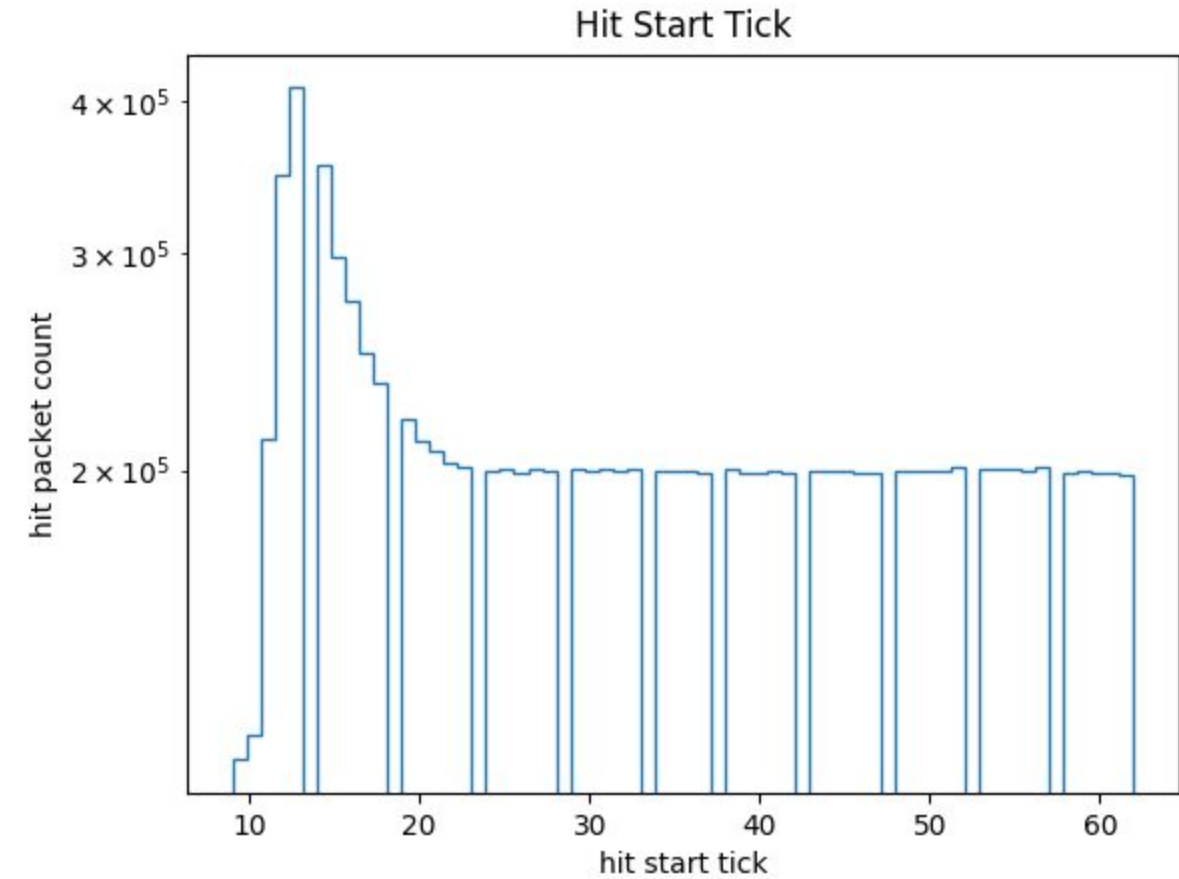
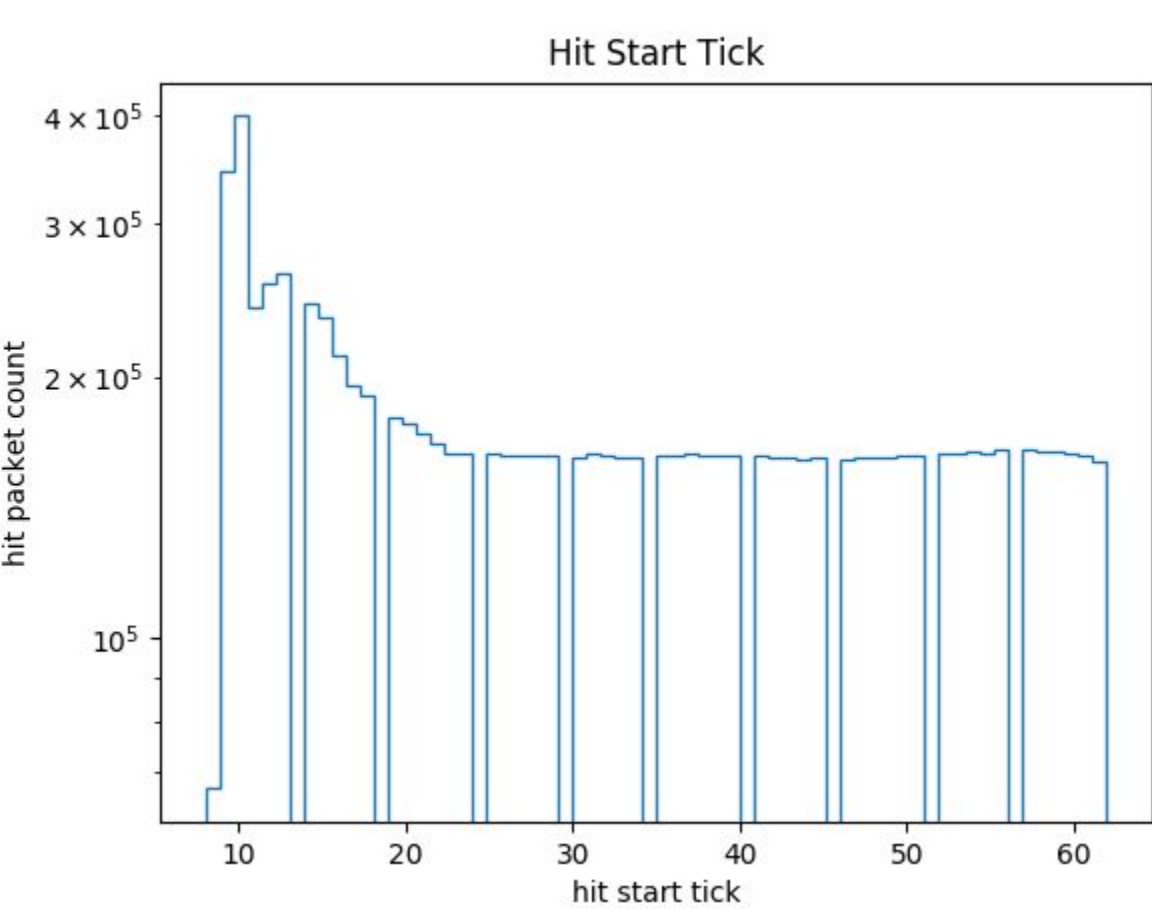
# Slots



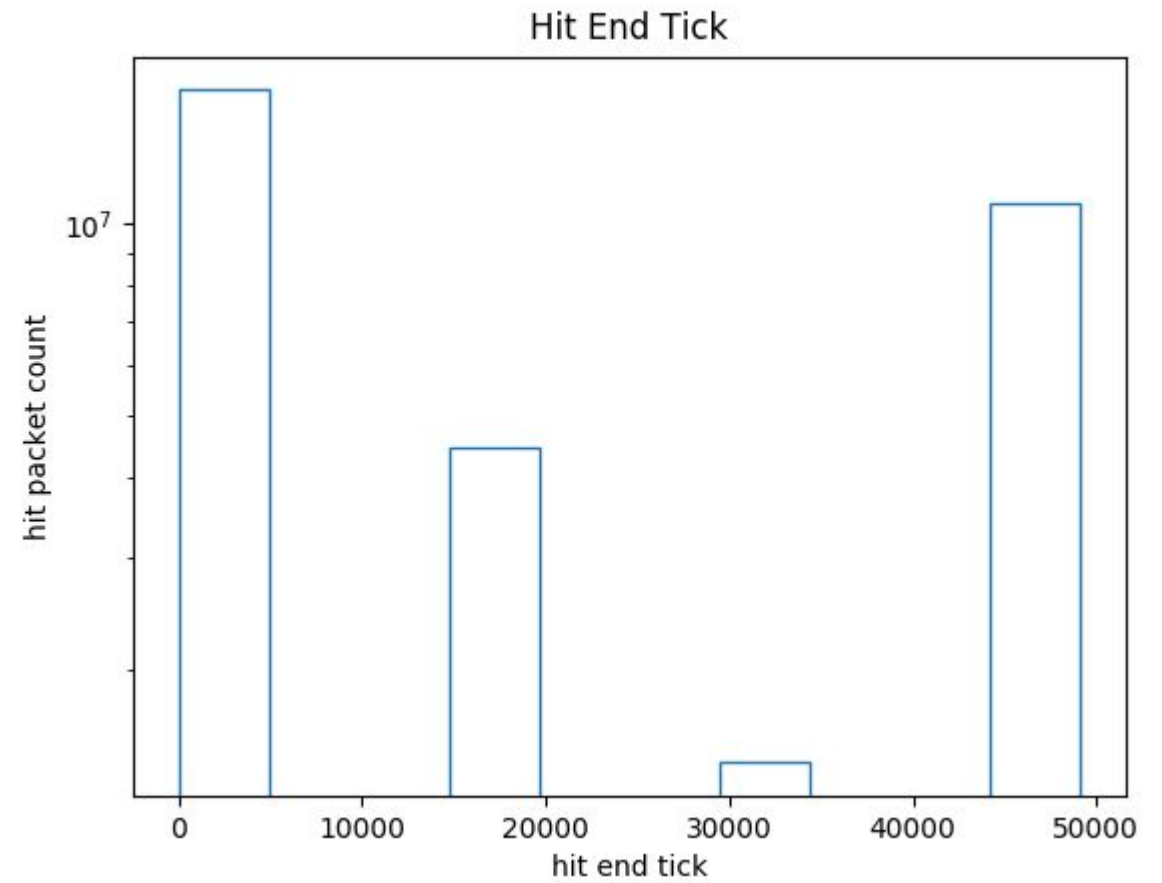
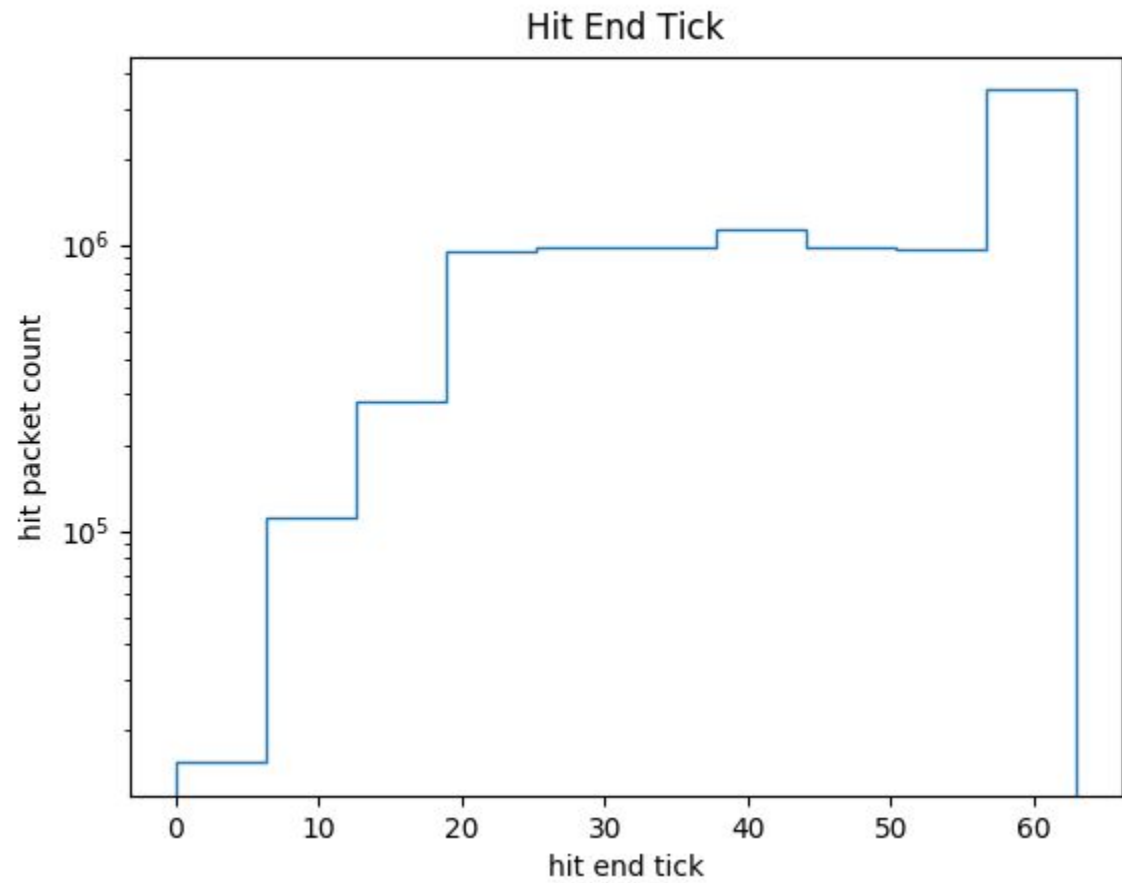
# Hit Start Tick



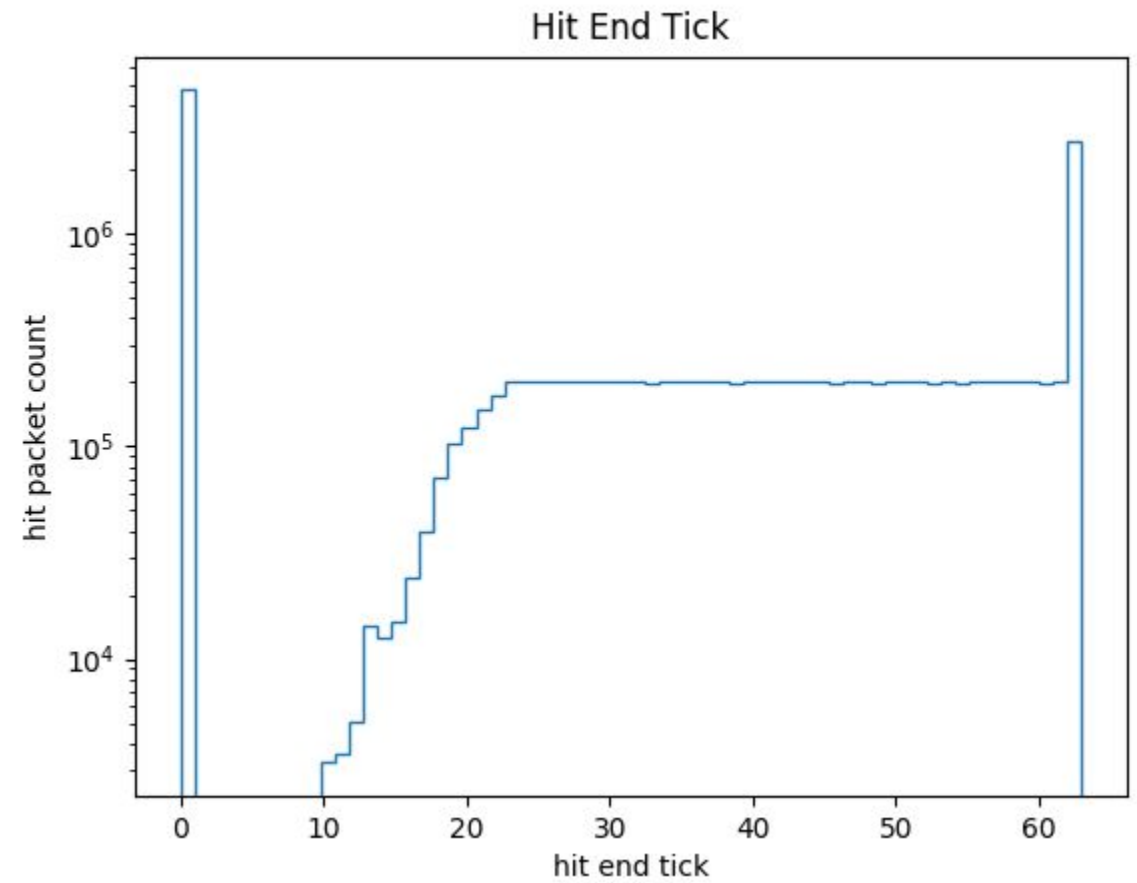
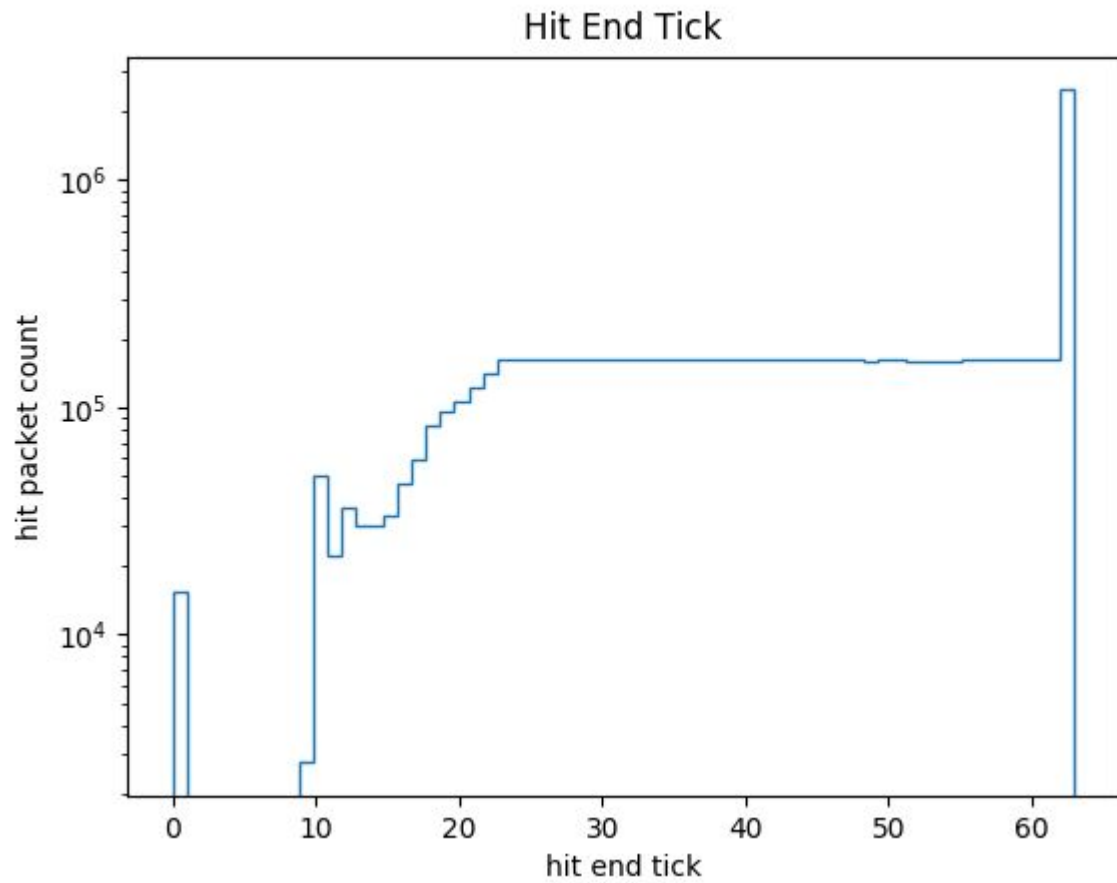
# Hit Start Tick (0-63)



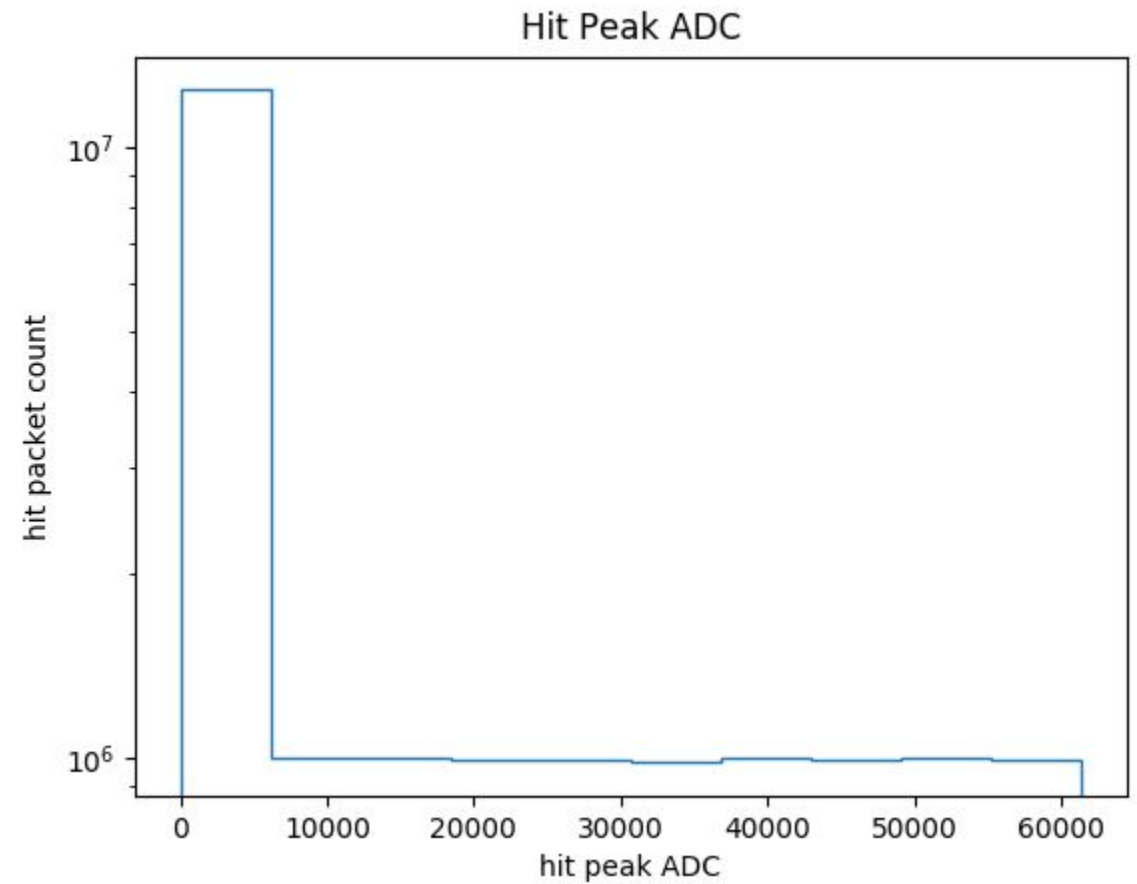
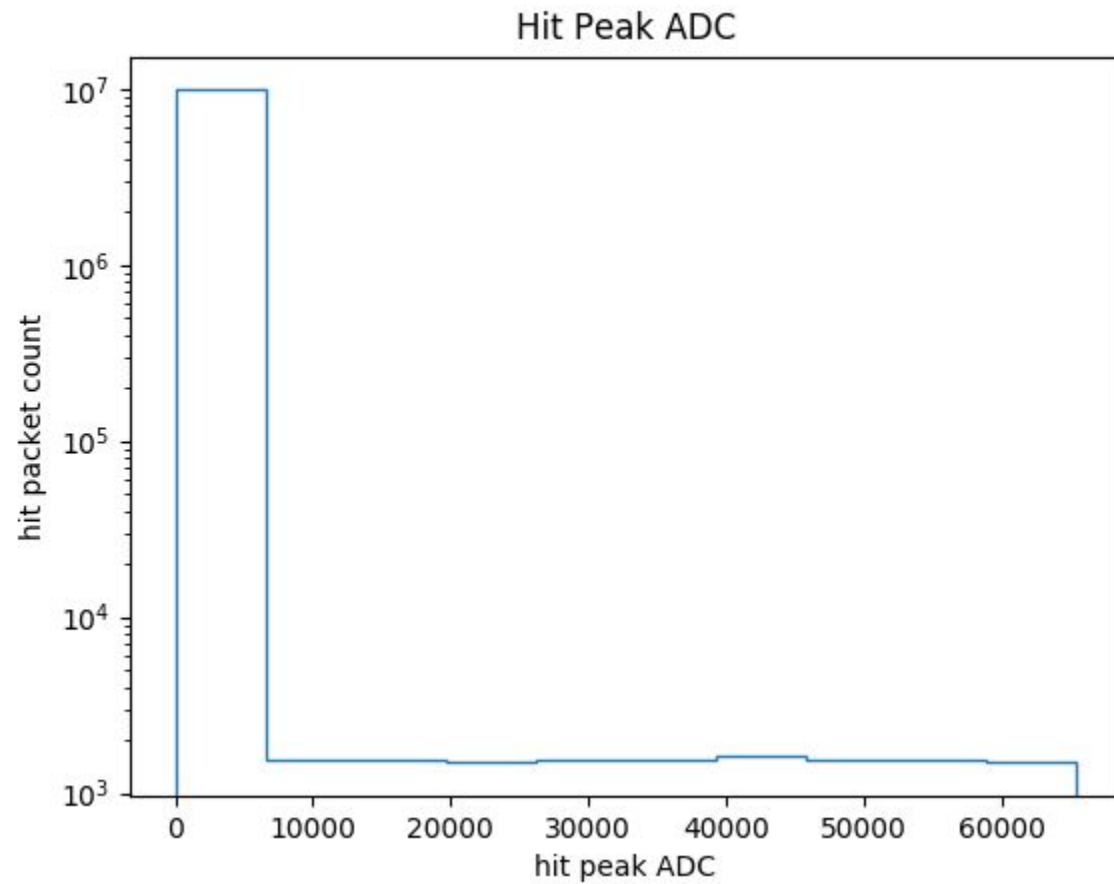
# Hit End Tick



# Hit End Tick (0-63)

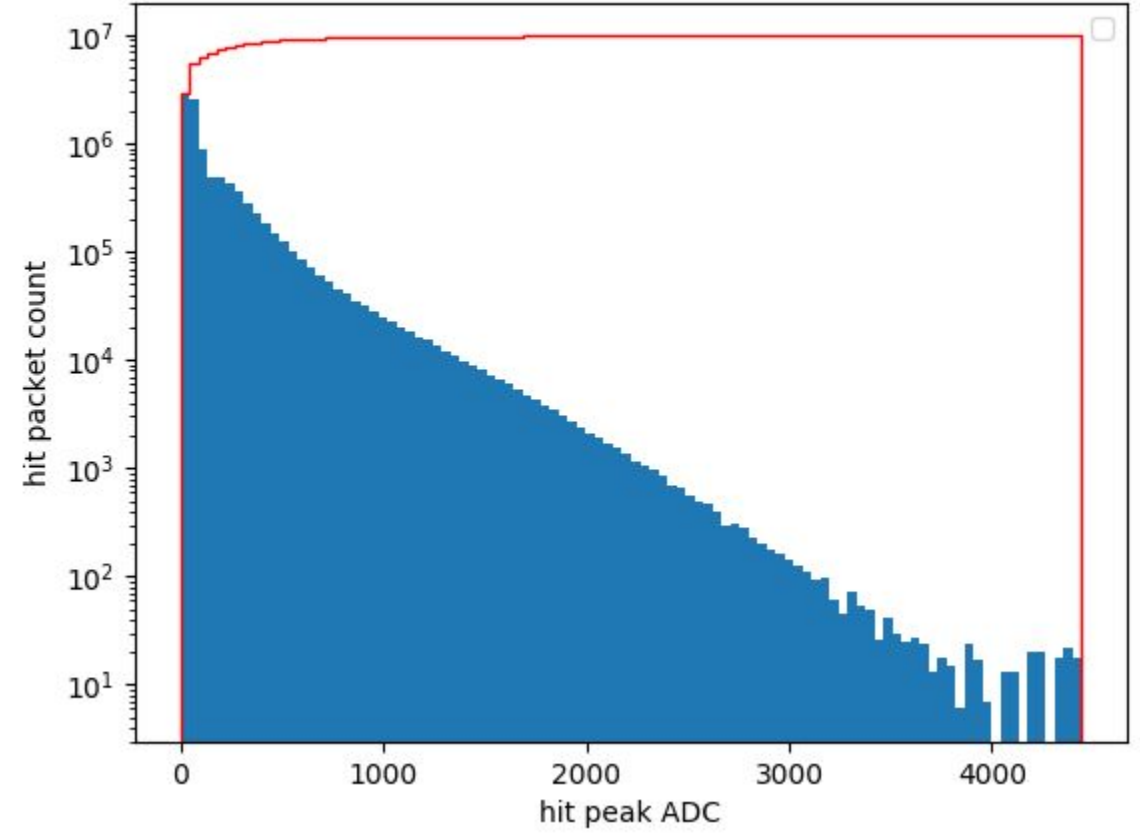


# Hit Peak

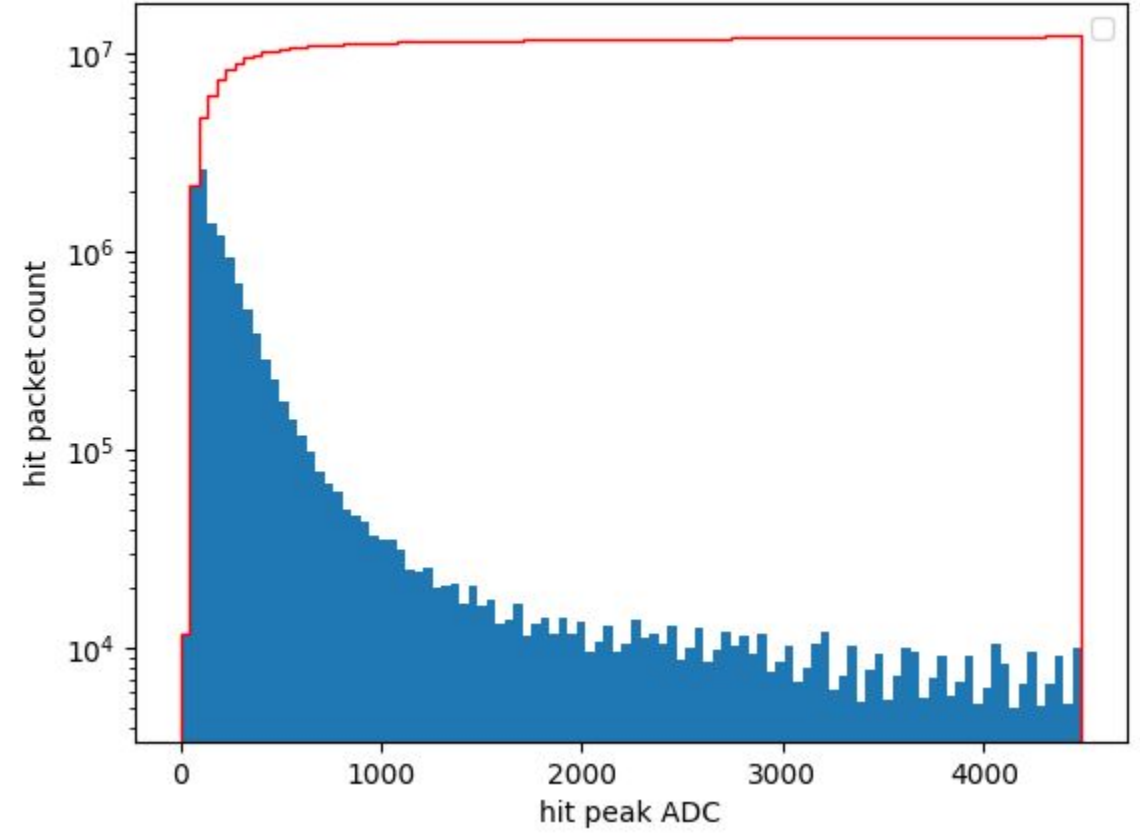


# Hit Peak (0-4492)

Hit Peak ADC up to 4492

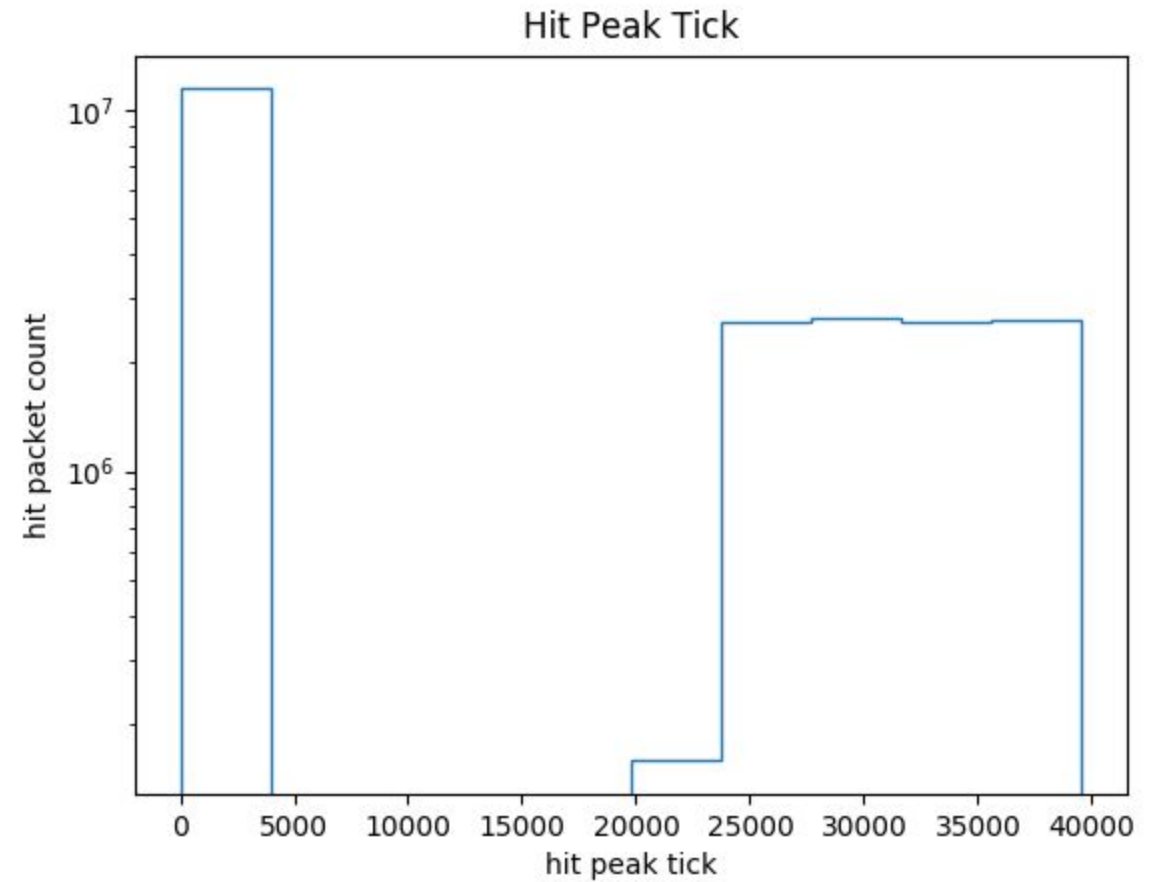
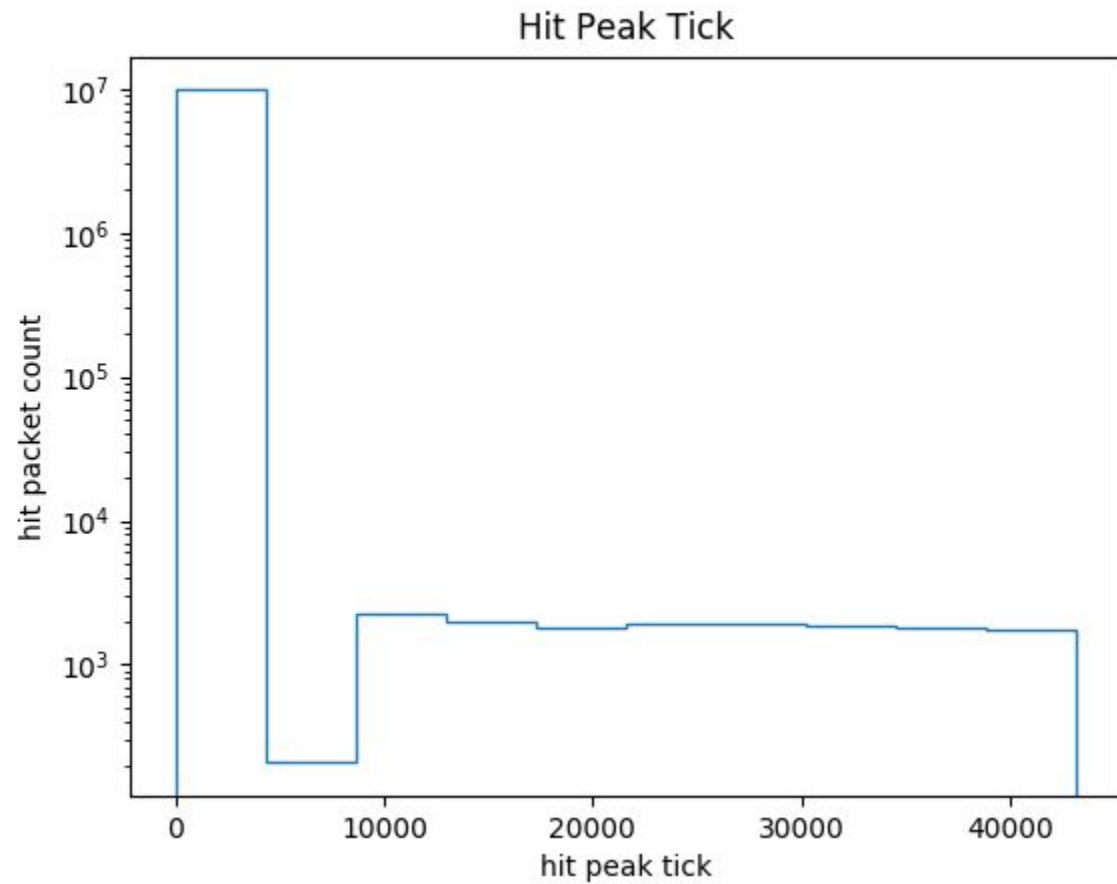


Hit Peak ADC up to 4492

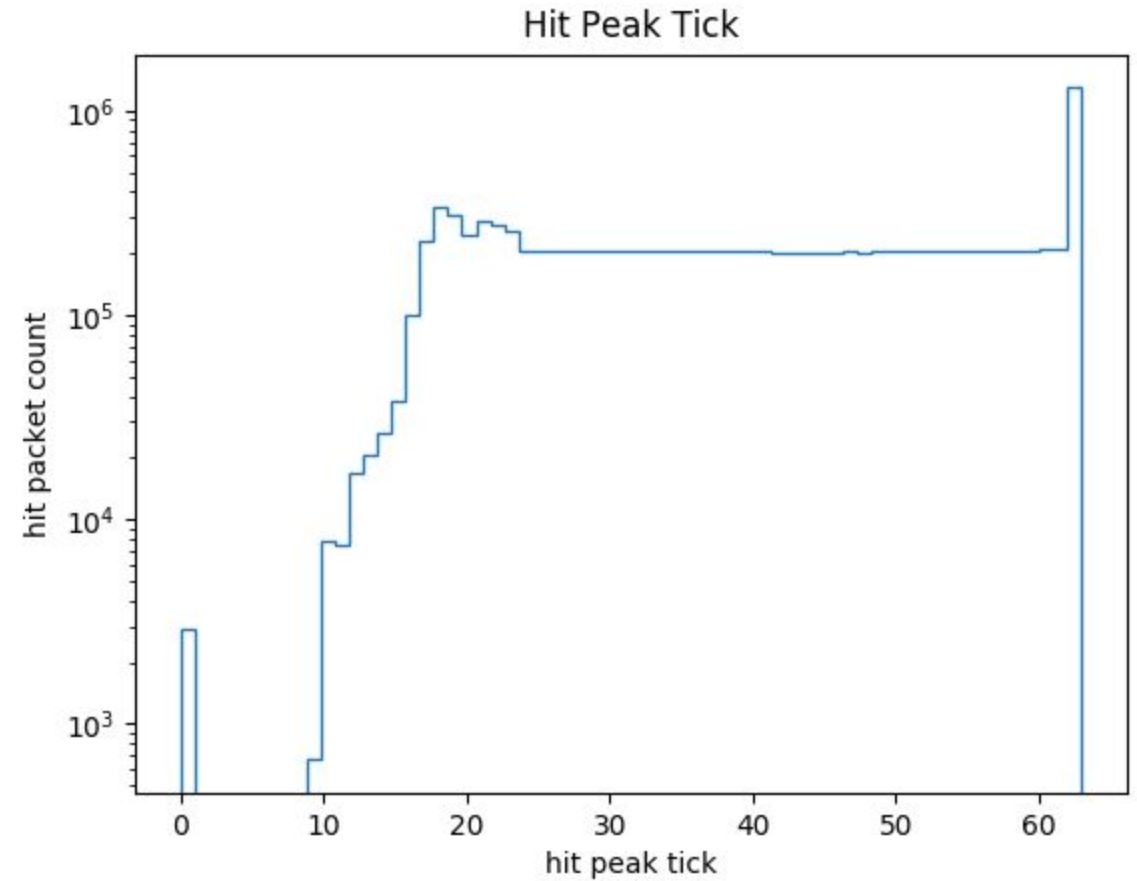
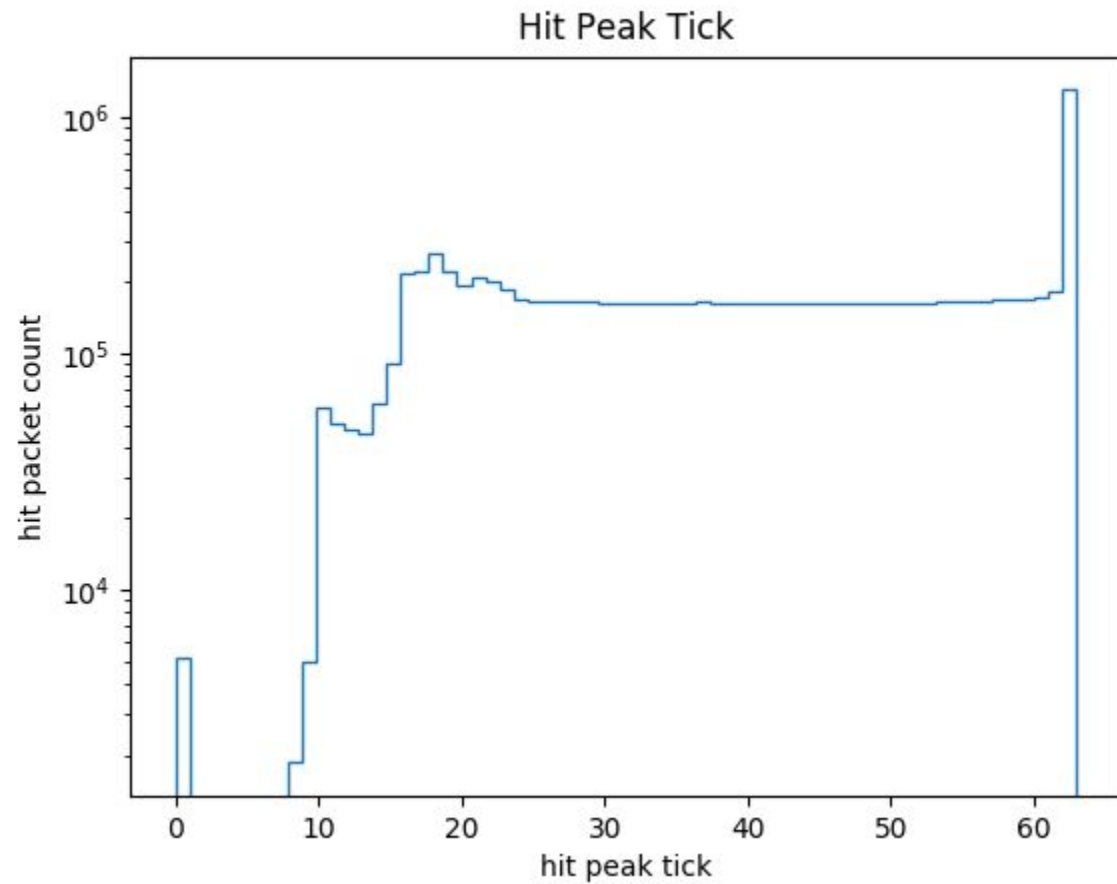




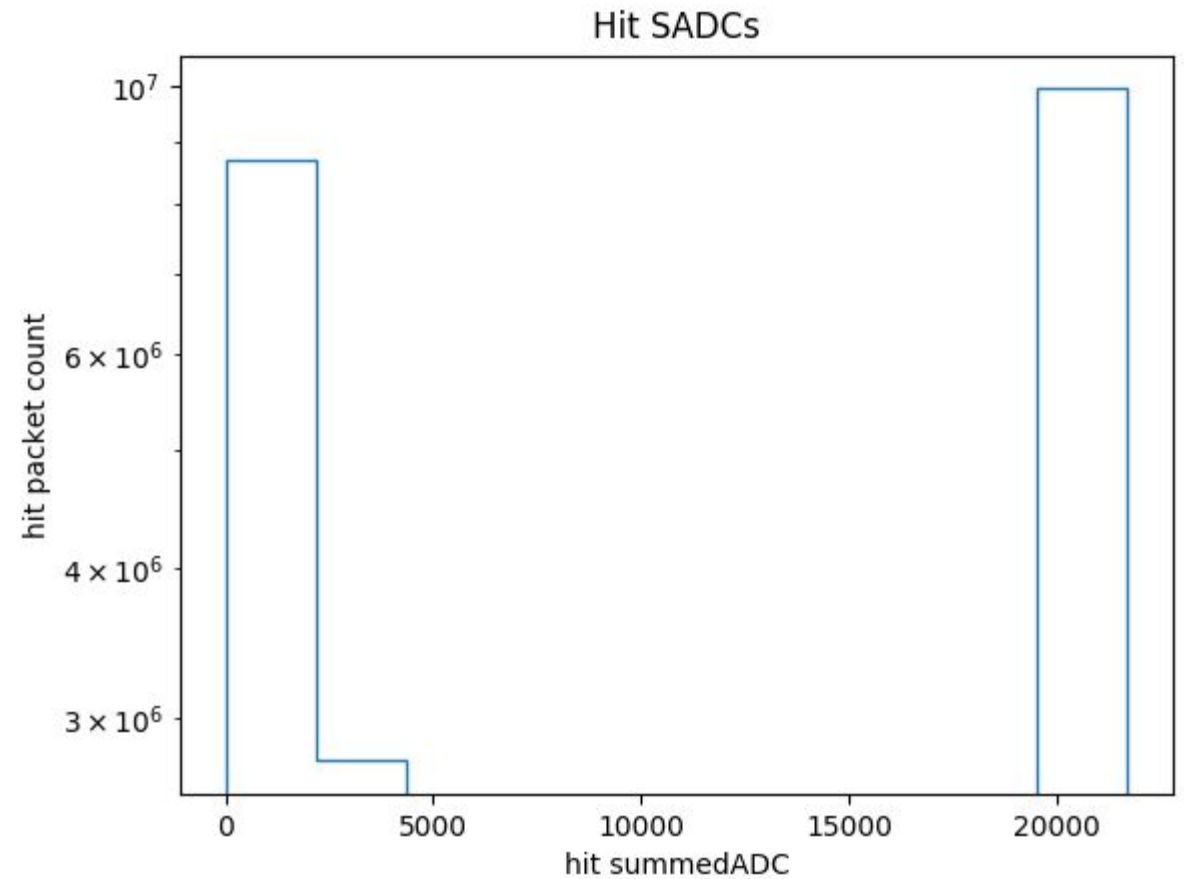
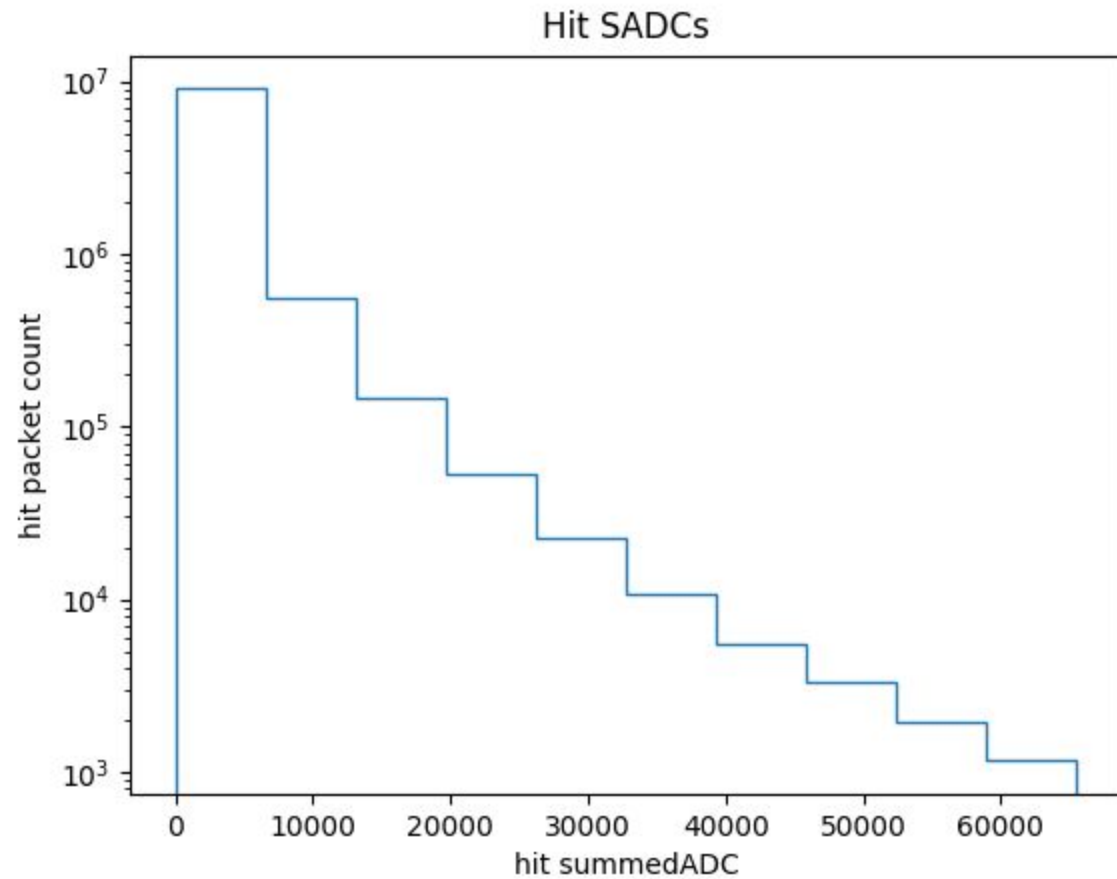
# Hit Peak Tick



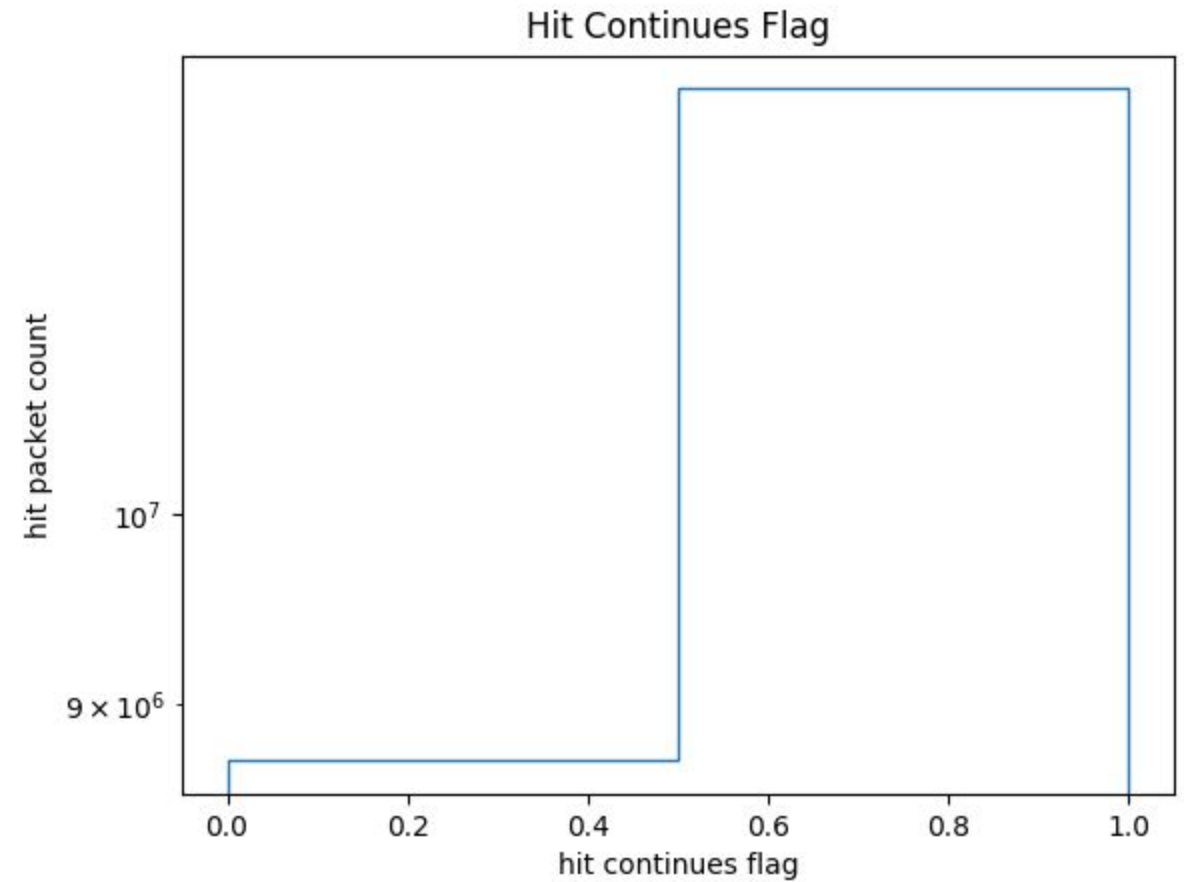
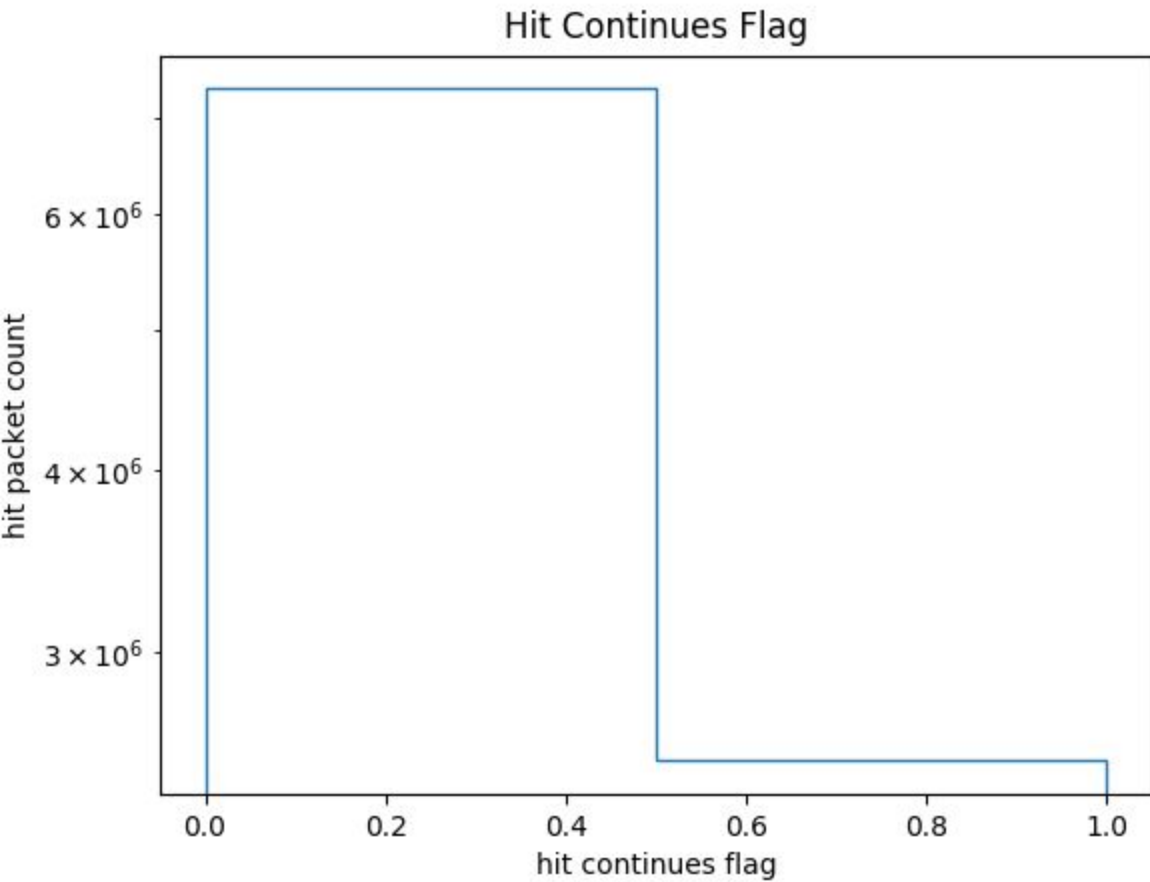
# Hit Peak Tick (0-63)



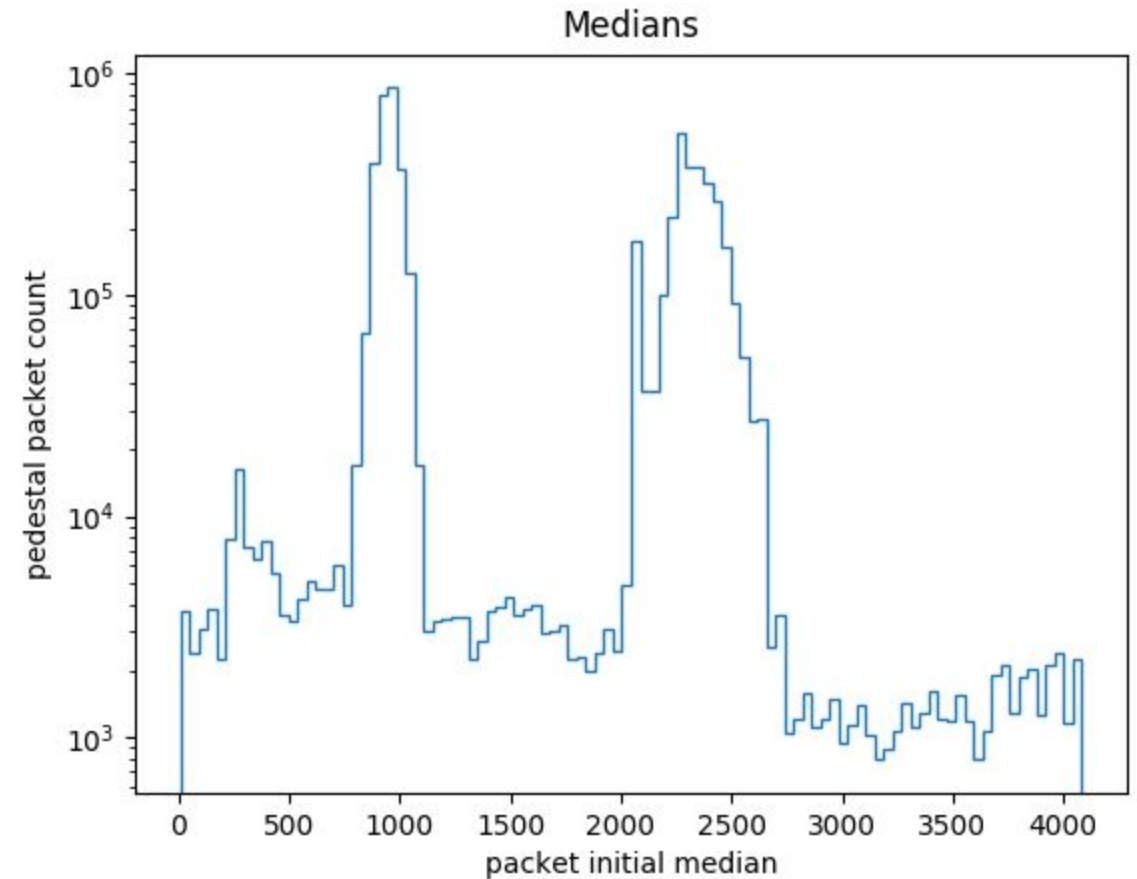
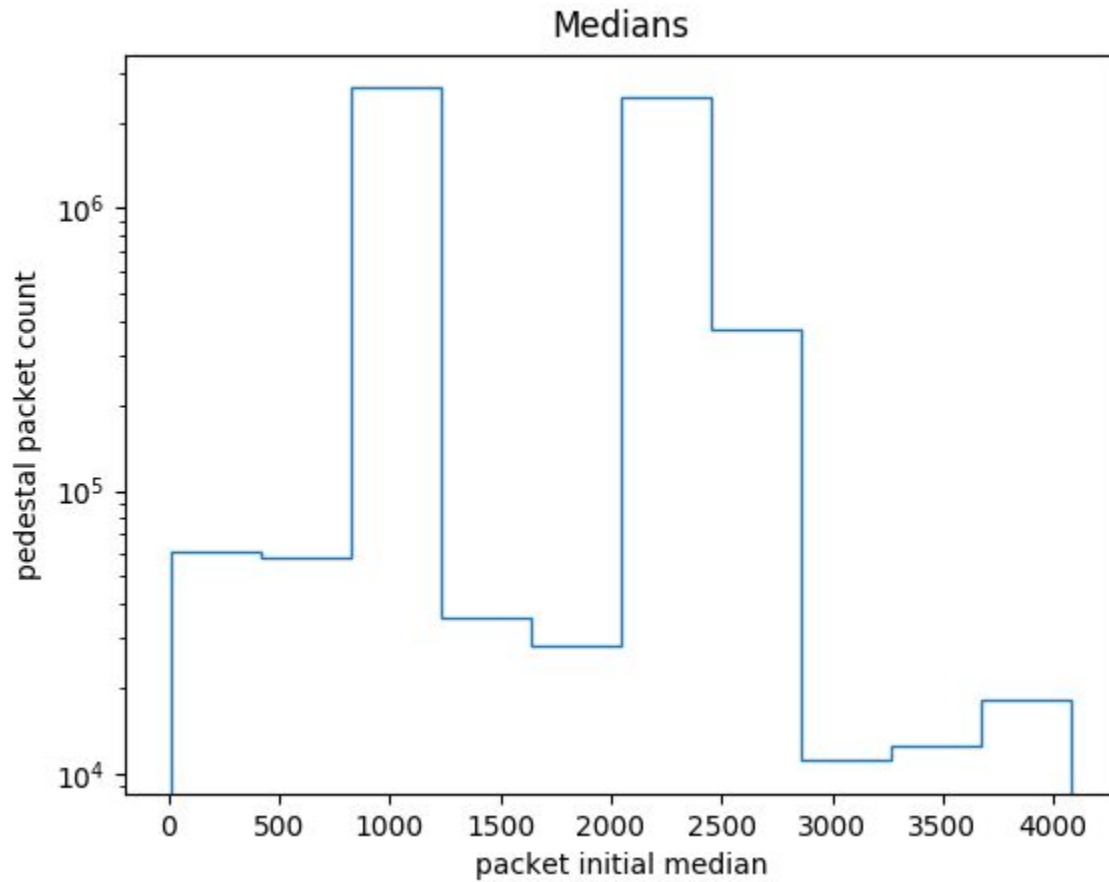
# Hit SummedADC



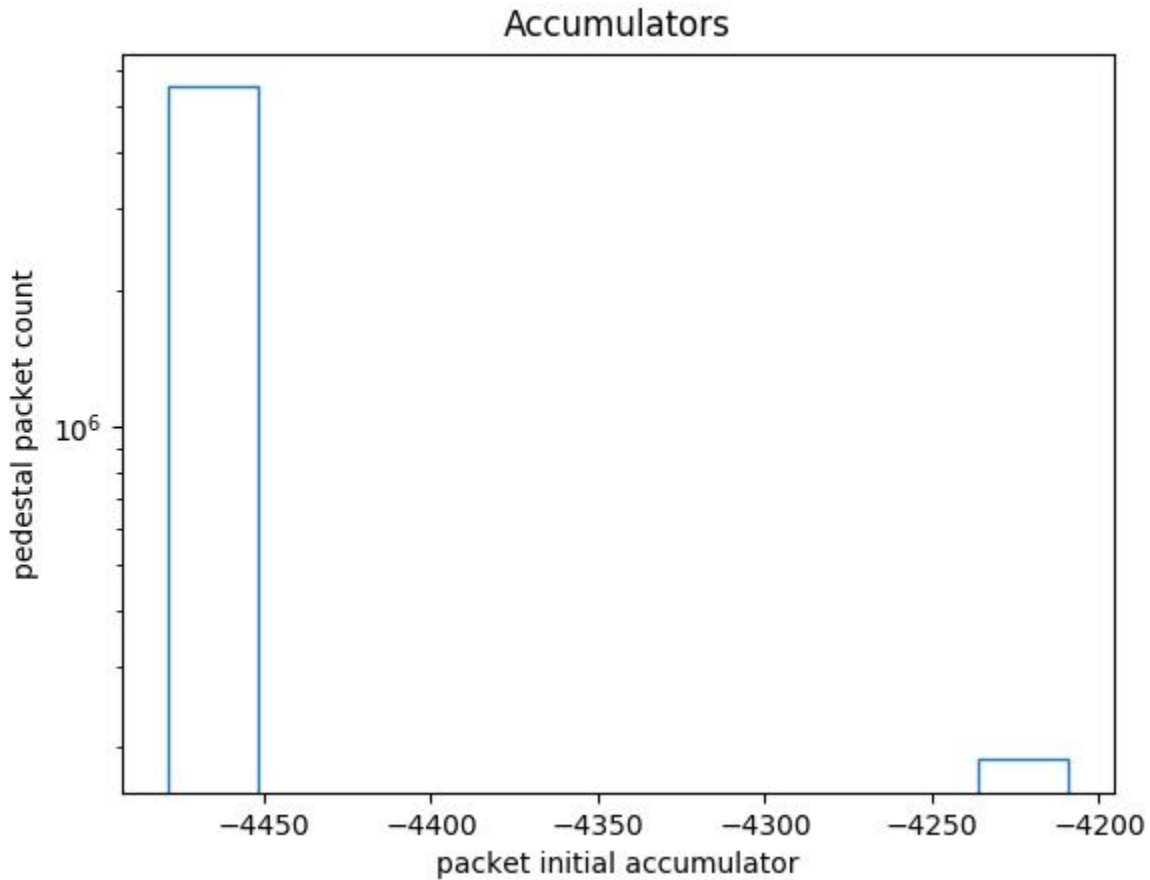
# Hit Continues Flag



# Packet Initial Median



# Packet Initial Accumulator



This plot was gotten by subtracting 4492 from anything beyond the expected positive range for the accumulator (0-10). The two values gained are in fact: 283 and 13

accumulators, entries:5718873, uniquevals:{283: 188421, 13: 5530452}

# Summary

---

- We know there is an issue in the firmware where it looks like either a header frame is flagged as a hit frame or a header frame and a hit frame are overlaid on top of each other. So some issues with hit quantities out of acceptable range might come from that.
- Beyond that I think there are some strange/unexplained plots for: fibreNos, crates, flags, slots, hitsummedADCs, hitContinues and accumulator. Not clear whether some outlying quantities are from crawlerscript/fw bugs yet.
- Need to be able to more reliably identify whether the quantities which must be bugs are a result of bugs in the firmware or in the crawler script I use to parse the binary data. I think it is worth replicating these plots using hit objects decoded via the hitBinaryMangler.py and hitFormatter.py dataflow-software tools to confirm